ARPA Network Information Center
Stanford Research Institute
Menlo Park, California 94025

F. Tobagi
L. Kleinrock

June 29, 1973

NIC # 17367
Packet Radio Temporary Note No. 67

Simulation of Various Channel Access Schemes

1. INTRODUCTION

The purpose of the simulation is two-fold:

1 - to check the validity of some of the assumptions made in the mathematical models, or to see the effect on the results due to the relaxation of these assumptions.

2 - to investigate various access schemes which appear interesting but hard to analyze.

In this note we concern ourselves only with the first objective, focusing on the Aloha access mode and the carrier sense access modes previously analyzed in PRT notes 24 and 37 [K1,K2].

In section 2, we discuss the results obtained from the simulation.

In section 3, we give a brief description and a rough algorithm of the simulator.

2. DISCUSSION

2.1 Background and previous assumptions

The access modes we are interested in at this point are the Aloha access mode and the carrier-sense access mode with or without initial random transmission delay (IRTD).

For the mathematical models discussed in previous Packet Radio Temporary notes [N1, K1, K2], the following assumptions were made:

A1 - There is an infinite number of users who collectively form a Poisson input source.

A2 - The retransmission delay introduced in case of a conflict is large compared to the transmission time of a packet.

A3 - The traffic entering the channel is an independent process, considered moreover to be Poisson.

Loosely speaking, A3 follows from assumption A2.

F. Tobagi
L. Kleinrock

June 29, 1973

Simulation of Various Channel Access Schemes

Under assumptions A1, A2 and A3, one could analytically find a relation between the channel traffic G and the throughput S, which we denote here by

$$S = f(G) \tag{1}$$

Conversely, for a given normalized input rate S, Eq. (1) is solved yielding a value for the normalized channel traffic G. An estimate for the average number of transmission and repetitions per packet was given by G/S.

After a transmission has taken place, a terminal has to wait a certain time before it claims the packet to be unsuccessful. Beyond that minimum waiting time required, an additional random waiting time X is introduced to avoid repeated conflicts.

If the system operation is such that positive acknowledgement packets are always correctly received with probability one (for example, by providing a separate channel for acknowledgement traffic), then the minimum waiting time can simply be equal to the acknowledgement packet's transmission time plus its propagation delay.

Under all these considerations, delay* equations were given and are repeated† here below:

Aloha:

$$D_{ALOHA} = \frac{G}{S}(T + \tau + T_a + \tau + \bar{X}) - \bar{X}$$
$$= \frac{G}{S}(2a + 1 + \alpha + \delta) - \delta \tag{2}$$

with $\delta = \frac{\bar{X}}{T}$.

Carrier-Sense I (without IRTD):

$$D_{C\text{-}S\text{-}I} = \frac{G}{S}(\bar{r}_1 + 2a + 1 + \alpha + \delta) - \delta \tag{3}$$

where $\bar{r}_1$ is the average pre-transmission delay.

Carrier-Sense with initial transmission delay:

$$D_{C\text{-}S\text{-}II} = \frac{G}{S}(a\bar{r}_2 + 2a + 1 + \alpha) \tag{4}$$

where $\bar{r}_2$ is here too an average pre-transmission delay and where it was claimed that

A4 - no waiting time $\bar{X}$ beyond the minimum required was necessary to avoid repeated conflicts.

---

* The delays considered here include the time of acknowledgement packets back to the terminal equal to $T_a + \tau$.

† The equations are normalized to T, the information packet's transmission time.

Equations (2) and (3) were computed assuming that

A5 - $\overline{X}$ (or δ) is independent of the load of the channel (it was arbitrarily chosen to be equal to 2).

2.2 The simulation model and its results

The simulation model assumes only the reasonable condition that the source is Poissior distributed (A1).

Major results

2.2.1 - For Aloha and Carrier-Sense I, it is impossible to achieve certain throughputs without relaxing assumption A5: $\overline{X}$ is a function of S; the higher the input rate is, the larger $\overline{X}$ must be to prevent a constantly increasing backlog, i.e., to prevent the channel from saturating.

Figure 1 compares simulation points to the analytical equation S=f(G) for carrier-sense I, a=0.01. The labels assigned to the simulation points correspond to the value of $X_{max}=2\overline{X}$ (2δ) which allows us to achieve such a throughput.

The simulation experiments were conducted such that the value $\overline{X}$ corresponds to achieving the indicated throughput with the minimum possible delay. However, due to time consumption, such an experimental design was not very finely conducted.

In Figure 2, we compare the following delays:

(i) Eq. (3) (mathematical prediction) with $\overline{X} = 2$
(ii) simulation points with $\overline{X}$ as a label
(iii) Eq. (3) with same $\overline{X}$ as for the simulation points.

It is obvious to note that the actual delays encountered are, for S > .2, larger than those provided by (i) (and considered in previous packet radio notes [N1, K1]).

But it is interesting to note that the points provided by (ii) and (iii) are satisfactorily close to each other. This leads us to believe that at the expense of some finite value of retranmission delay (relaxing A2), assumption A3 is not unreasonable.

The same work was performed for the Pure Aloha access mode. The simulation results* are summarized in Table 1.

---

* In this part, delays do not include the time for positive acknowledgement.

Figure 3 is a plot of the various values of X(S) considered in the simulation. In an attempt to find an expression for X(S), a piece-wise linear approximation is given on the semi-logarithmic graph. That is, X(S) satisfies the following equation:

$$X(S) = \begin{cases} 4 \times 2^{S/.05} & S < .05 \\ 8 \times \left(5.5\right)^{\frac{S-0.5}{.09}} & 0.05 < S < 0.14 \\ 44 \times \left(\frac{25}{11}\right)^{\frac{S-.14}{.02}} & .14 < S < .16 \\ 100 \times \left(3\right)^{\frac{S-.16}{.02}} & .16 < S < .18 \\ \infty & S > .18 \end{cases} \qquad (5)$$

In Fig. 4, we compare simulation results to those obtained by Eq. (5) and

$$\Gamma_{ALOHA}(S) = (e^{2G} - 1)\left(2 + \frac{X(S)}{2}\right) + 1 \qquad (6)$$

2.2.2 Carrier-sense with IRTD

Recall that, in this scheme, when a terminal does not receive a positive acknowledgement within $T_a + \tau$ sec. following the transmission of its packet, or if a terminal has lost its chance (because of the initial delay due to p), then the packet is ready for retransmission. Obviously, such operating mode does not satisfy the major assumptions made in the analysis, namely A2 and A3.

The accumulation of packets ready for transmission at the end of a transmission period constitutes what we call "bunching."

The bunching effect was noticed in the following observations:

a) The maximum throughput attainable (before reaching saturation, i.e., before the existence of a constantly increasing backlog) is lower than predicted by the analysis.

For example, for a=0.01, we have

p	maximum attainable	maximum predicted
0.1	0.5	0.79
0.6	0.3	0.6

Figure 3 is a plot of the various values of $X(S)$. considered in the simulation. In an attempt to find an expression for $X(S)$, a piece-wise linear approximation is given on the semi-logarithmic graph. That is, $X(S)$ satisfies the following equation:

$$X(S) = \begin{cases} 4 \times 2^{S/.05} & S < .05 \\ 8 \times (5.5)^{\frac{S-0.5}{.09}} & 0.05 < S < 0.14 \\ 44 \times \left(\frac{25}{11}\right)^{\frac{S-.14}{.02}} & .14 < S < .16 \\ 100 \times (5)^{\frac{S-.16}{.02}} & .16 < S < .18 \\ \infty & S > .18 \end{cases} \qquad (5)$$

Digitized by the Internet Archive
in 2025 with funding from
Amateur Radio Digital Communications, Grant 151

https://archive.org/details/temporary-note-067

b) For an achievable input rate, the corresponding channel traffic, and hence the delay, are larger than predicted.

That is, the bunching effect is to reduce the probability of success of a packet, increasing the traffic level and therefore reaching saturation for a much lower input rate.

It is to be noted, however, that assumption A4 is true.

A good operation of the scheme can be obtained by breaking up the bunching effect. That is, the system tends to satisfy assumptions A2 and A3 if we introduce a "spreading factor" X_s over which we randomize the rescheduling of a packet which either has failed to be transmitted or was unsuccessful. More precisely, when a terminal learns about its unsuccessful packet, it reschedules it for a uniformly chosen time over an X_s period. When a terminal fails to transmit, it behaves as if it has conflicted, that is, it delays the scheduling of the packet by uniformly chosen time between $T+T_a+\tau$ and $T+T_a+\tau+X_s$.

Observations:

1) The larger X_s is, the higher is the maximum throughput attainable, and the closer can one get to the function $S=f(G)$.

2) For a given throughput S, the smallest value for X_s does not necessarily provide the lowest delay. Quite a few experiments were conducted in an attempt to find the optimum X_s.

3) For a given value of S, the smaller p is, the smaller is the optimum X_s.

Results:

The simulation results for various values of a and p are summarized in Figures 5 to 13. The labels assigned to the simulation points represent the corresponding values of $X_s=2\bar{X}_s$. We also note in Fig. 9 that the inclusion of $\bar{X}_s$ into Eq. (4) to yield

$$D_{C\text{-}S\text{-}II} = \frac{G}{S}[a\bar{r}_2 + 2a + 1 + \alpha] + (\frac{G}{S} - 1)\bar{X}_s \qquad (7)$$

gives delay estimates which agree well enough with the simulation results. This leads us, here too, to believe that, at the expense of relatively small values of X_s, one can achieve delays and throughputs in the same order of magnitude as those provided by the analysis under assumptions A2 and A3.

TABLE 1

Simulation Results for Aloha

S	X	SIMULATION LENGTH IN (T)	AVERAGE NUMBER OF TRANS-MISSIONS	G	DELAY	AVERAGE
.05	4	8000	1.46	.0698	2.9	
.05	4	20000	1.2	.06	1.8	2.4
.05	5	20000	1.37	.068	2.49	
.05	6	9000	1.177	.056	1.9	
.05	6	10000	1.1	.052	1.48	1.86
.05	6	20000	1.208	.059	2	
.05	6	20000	1.25	.06	2.07	
.05	8	5000	1.147	.055	1.97	
.05	8	9000	1.179	.059	2.12	2.04
.05	8	20000	1.15	.058	1.93	
.05	8	20000	1.18	.0594	2.13	
.08	10	80000	1.31	.105	3.198	
.08	10	80000	1.314	.106	3.164	3.38
.08	10	80000	1.37	.109	3.585	
.08	14	80000	1.286	.104	3.64	
.08	14	80000	1.28	.103	3.47	3.56
.08	14	80000	1.28	.104	3.57	
.1	14	5000	1.345	.137	4.192	
.1	14	20000	1.39	.136	4.6	4.38
.1	14	20000	1.37	.136	4.34	
.1	16	20000	1.42	.141	5.3	
.1	16	20000	1.33	.135	4.06	4.64
.1	16	20000	1.35	.133	4.57	
.1	18	20000	1.4	.139	5.4	
.1	18	20000	1.35	.137	4.8	5.2
.1	18	20000	1.4	.136	5.5	
.12	26	10000	1.52	.19	8.69	
.12	26	20000	1.56	.191	9.49	9
.12	26	20000	1.51	.184	8.69	
.12	28	20000	1.52	.18	9.36	
.12	28	20000	1.5	.184	8.85	9.1
.12	28	20000	1.52	.184	9.3	
.12	30	20000	1.54	.18	10.14	

(cont'd.)

TABLE 1 (cont'd.)

S	k	SIMULATION LENGTH IN (T)	AVERAGE NUMBER OF TRANS-MISSIONS	G	DELAY	AVERAGE
.14	40	20000	1.57	.213	13.14	15.17
.14	40	20000	1.61	.224	14.43	
.14	40	20000	1.64	.238	15.05	
.14	40	20000	1.62	.224	14.7	
.14	40	20000	1.75	.25	17.35	
.14	40	20000	1.70	.23	16.37	
.14	46	20000	1.638	.23	16.8	15.6
.14	46	20000	1.54	.206	14.14	
.14	46	20000	1.59	.225	16	
.16	100	20000	1.738	.276	39.8	40
.16	100	20000	1.75	.278	40	

K&E 10 X 10 TO THE INCH 46 0700
7 X 10 INCHES MADE IN U.S.A.
KEUFFEL & ESSER CO.

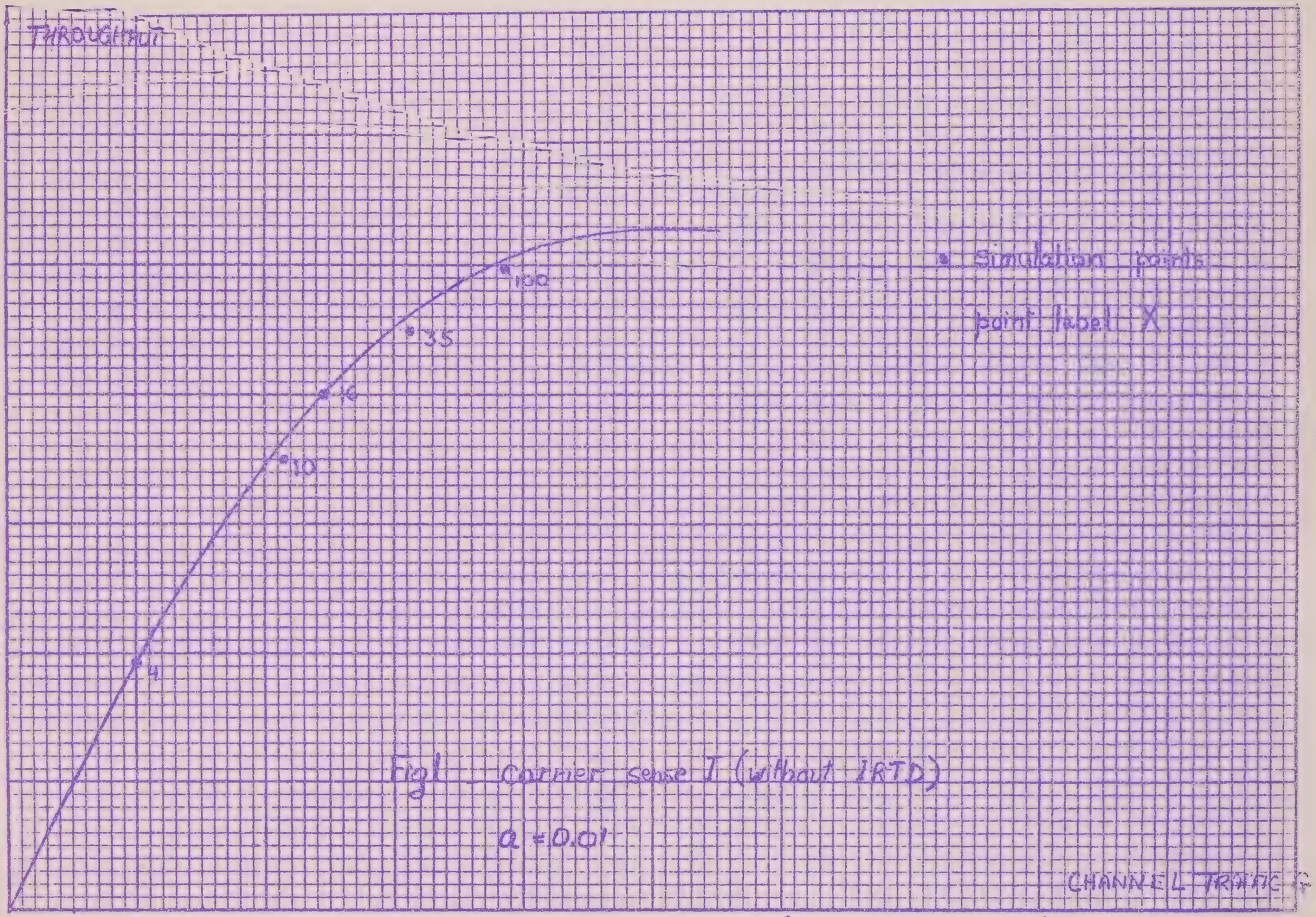
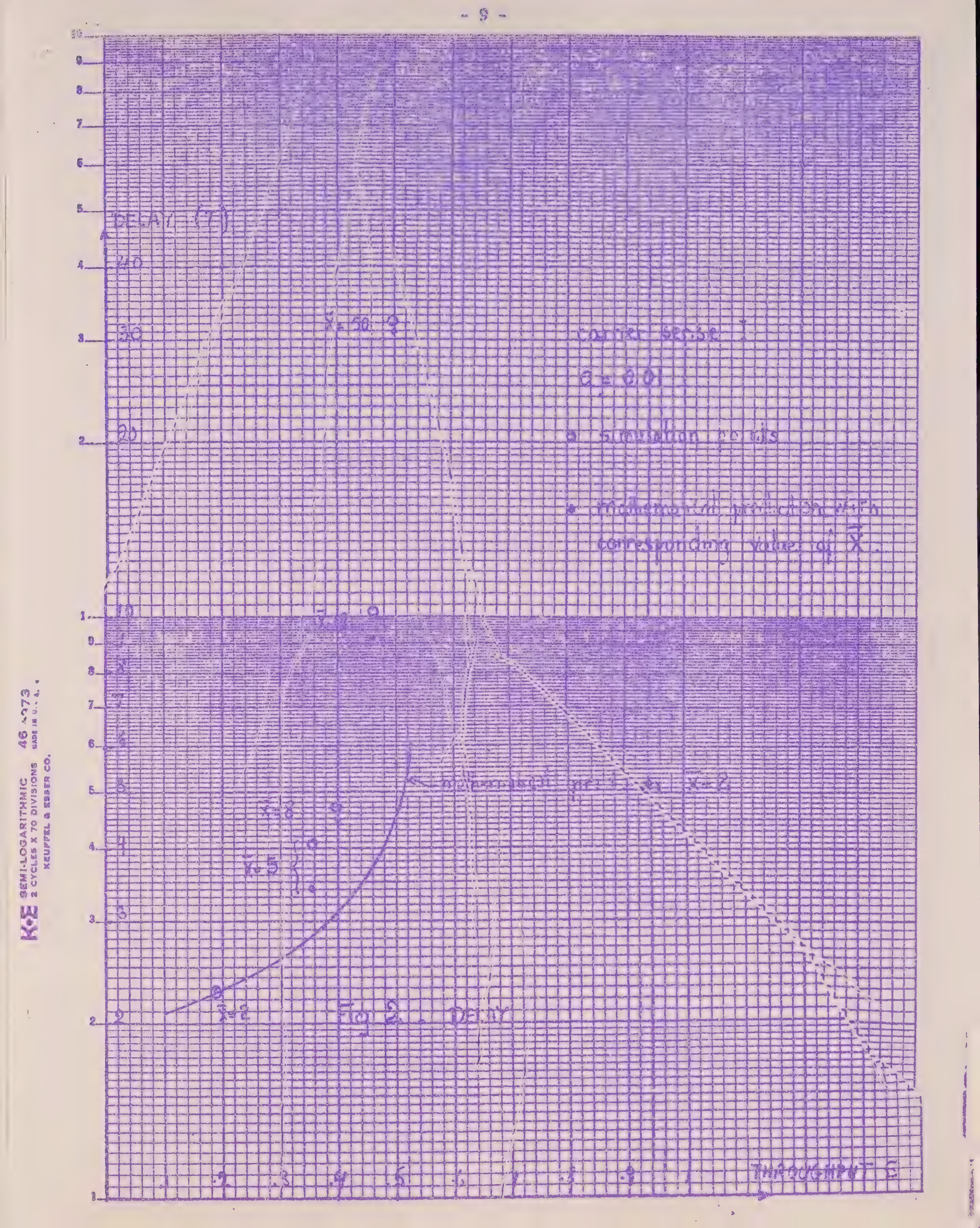
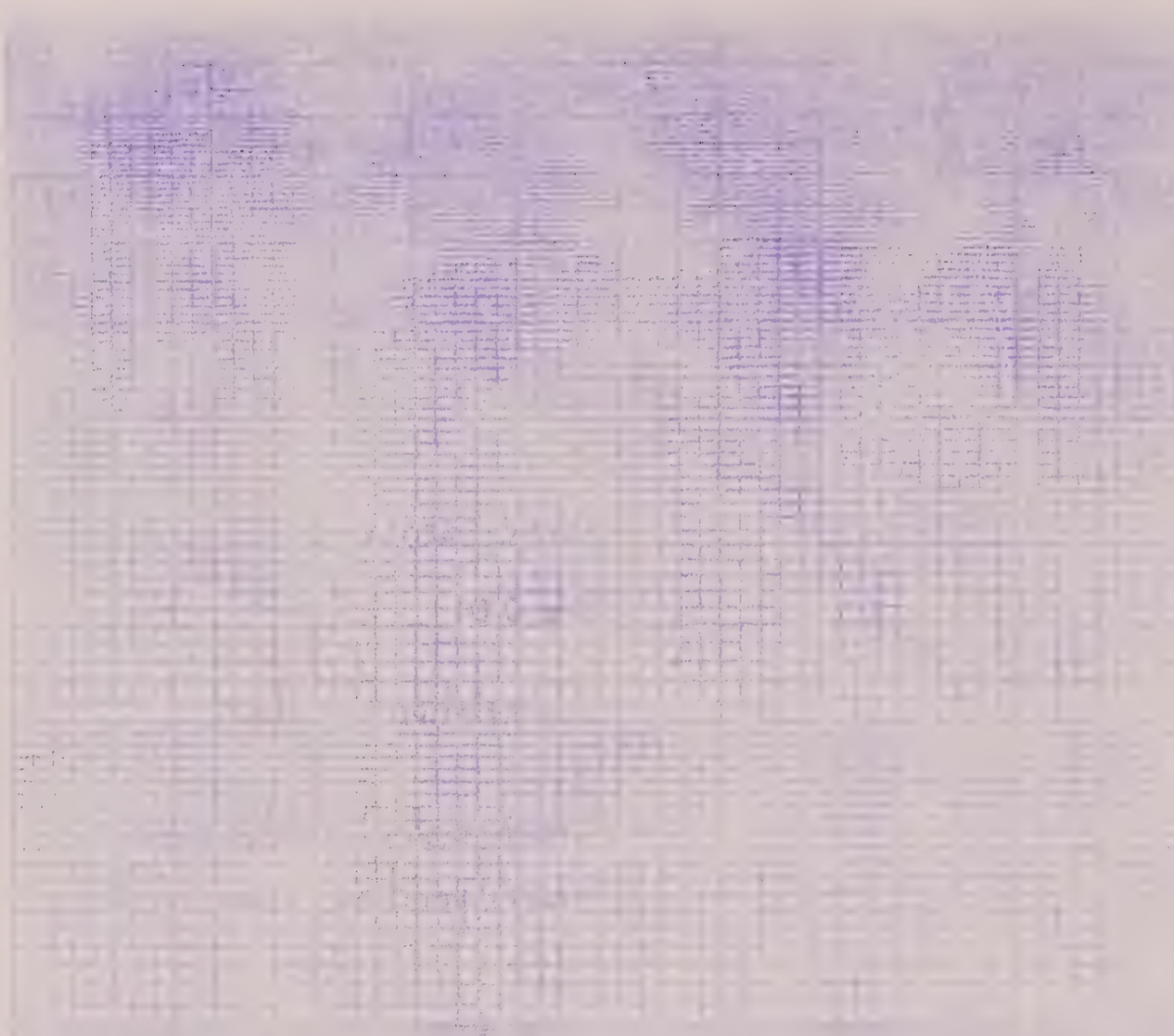


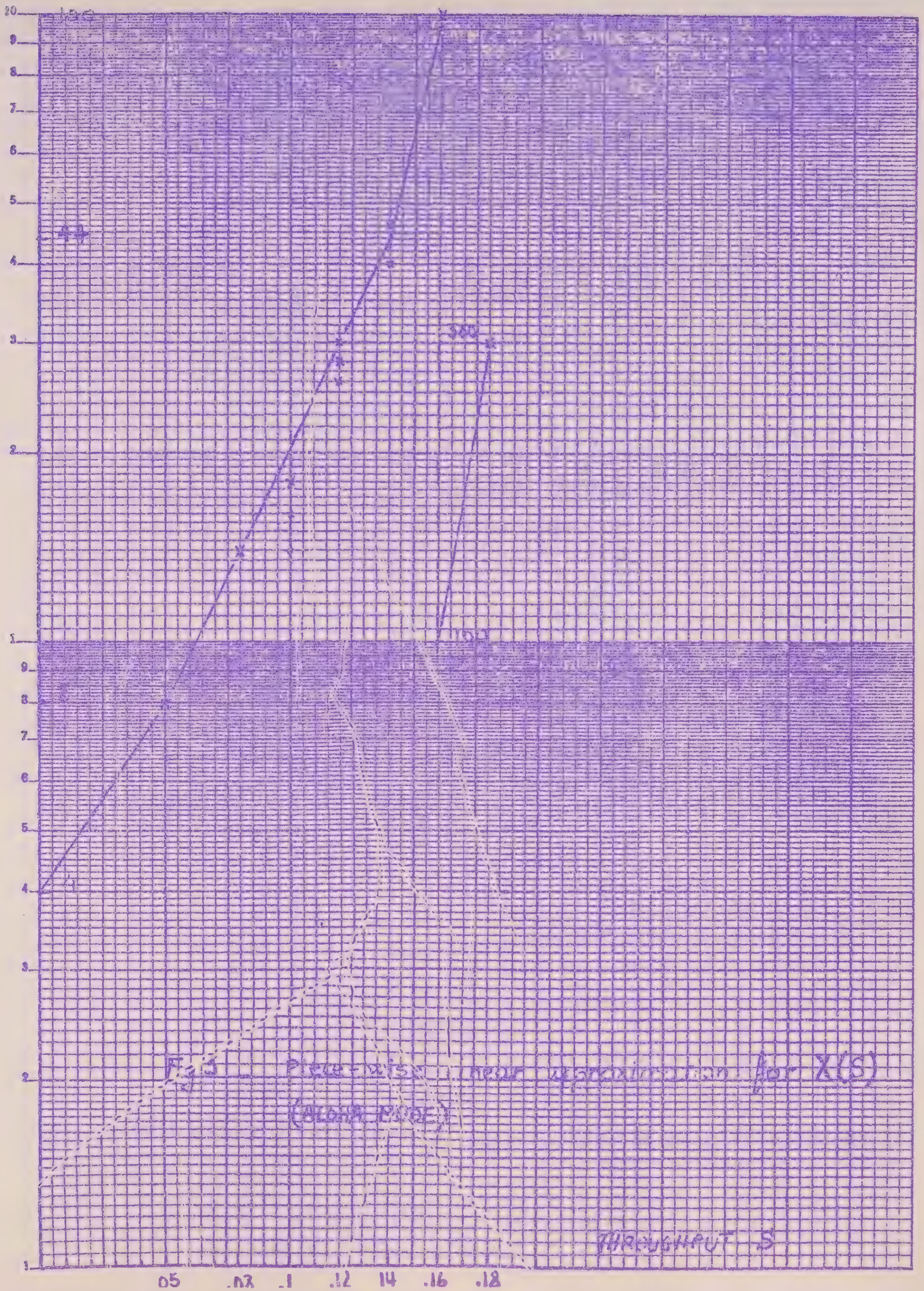
Fig 1 Corner sense I (without IRTD)

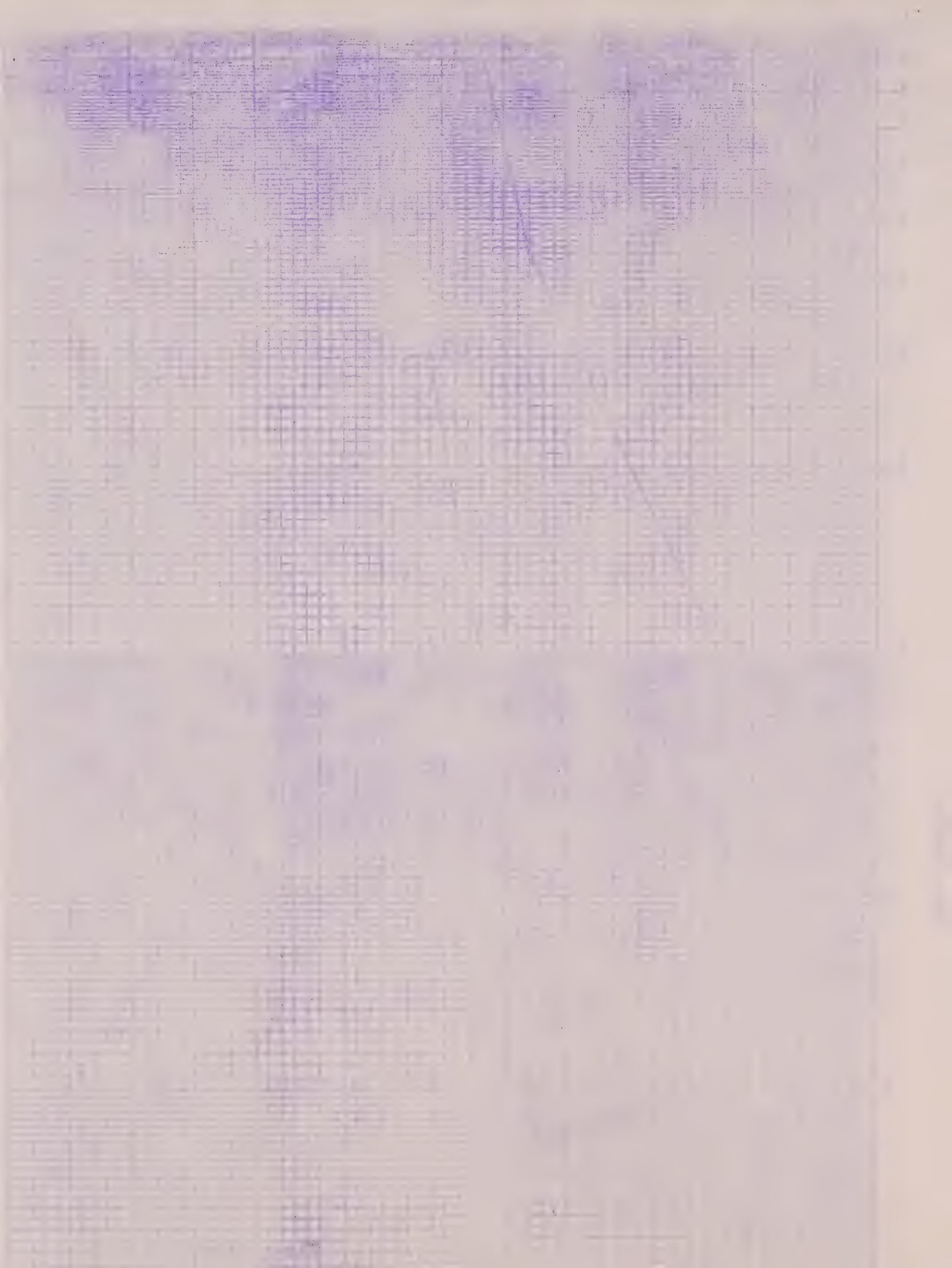
$a = 0.01$

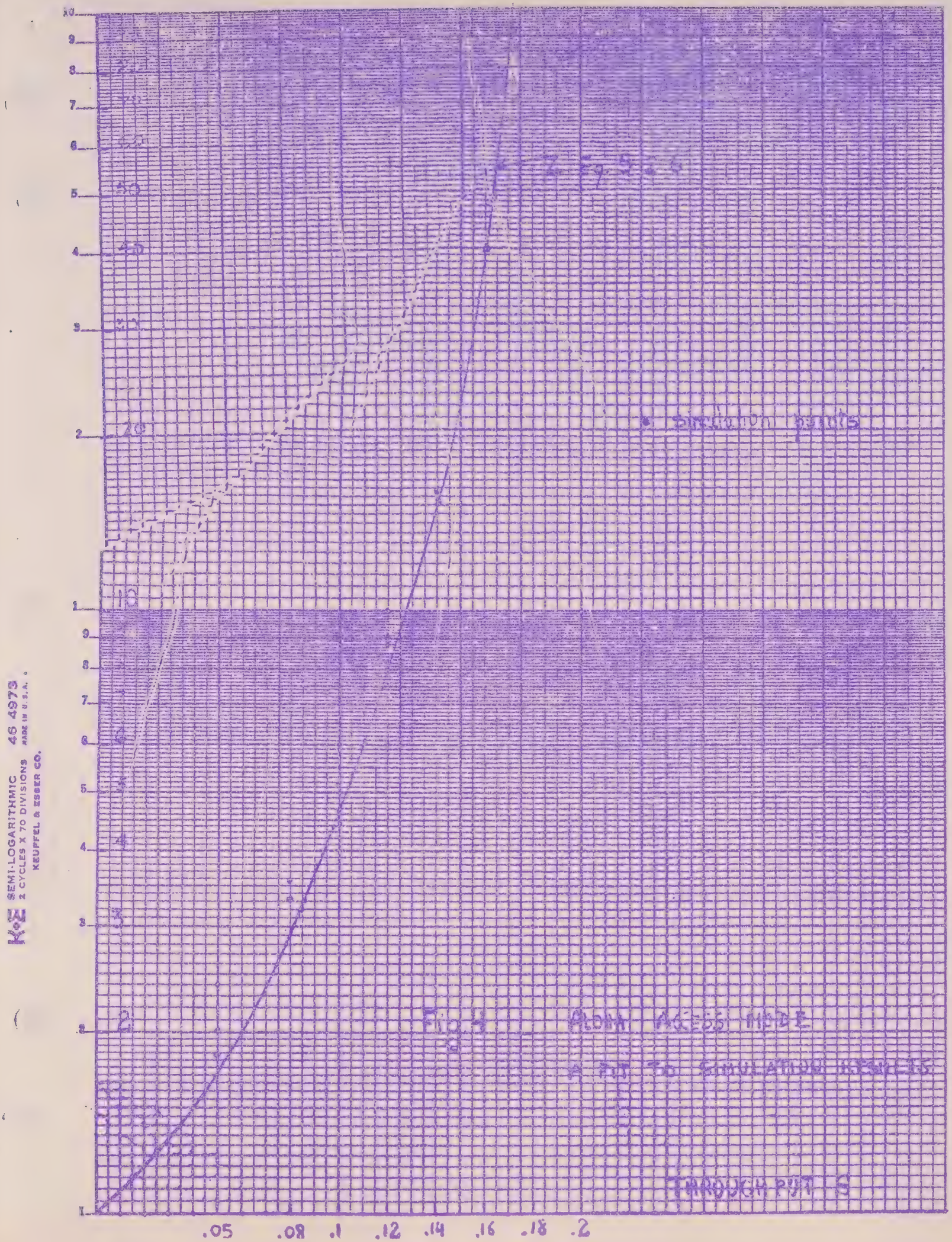


K+E SEMI-LOGARITHMIC 46 4273
2 CYCLES X 70 DIVISIONS
KEUFFEL & ESSER CO.




300
100
Fig 3 Piece-wise linear approximation for X(S)
(ALPHA EDGE)
THROUGHPUT S
.05 .08 .1 .12 .14 .16 .18





K+E SEMI-LOGARITHMIC 46 4973
2 CYCLES X 70 DIVISIONS MADE IN U.S.A.
KEUFFEL & ESSER CO.

K·E 5 X 5 TO ½ INCH 46 0860
7 X 10 INCHES MADE IN U.S.A.
KEUFFEL & ESSER CO.

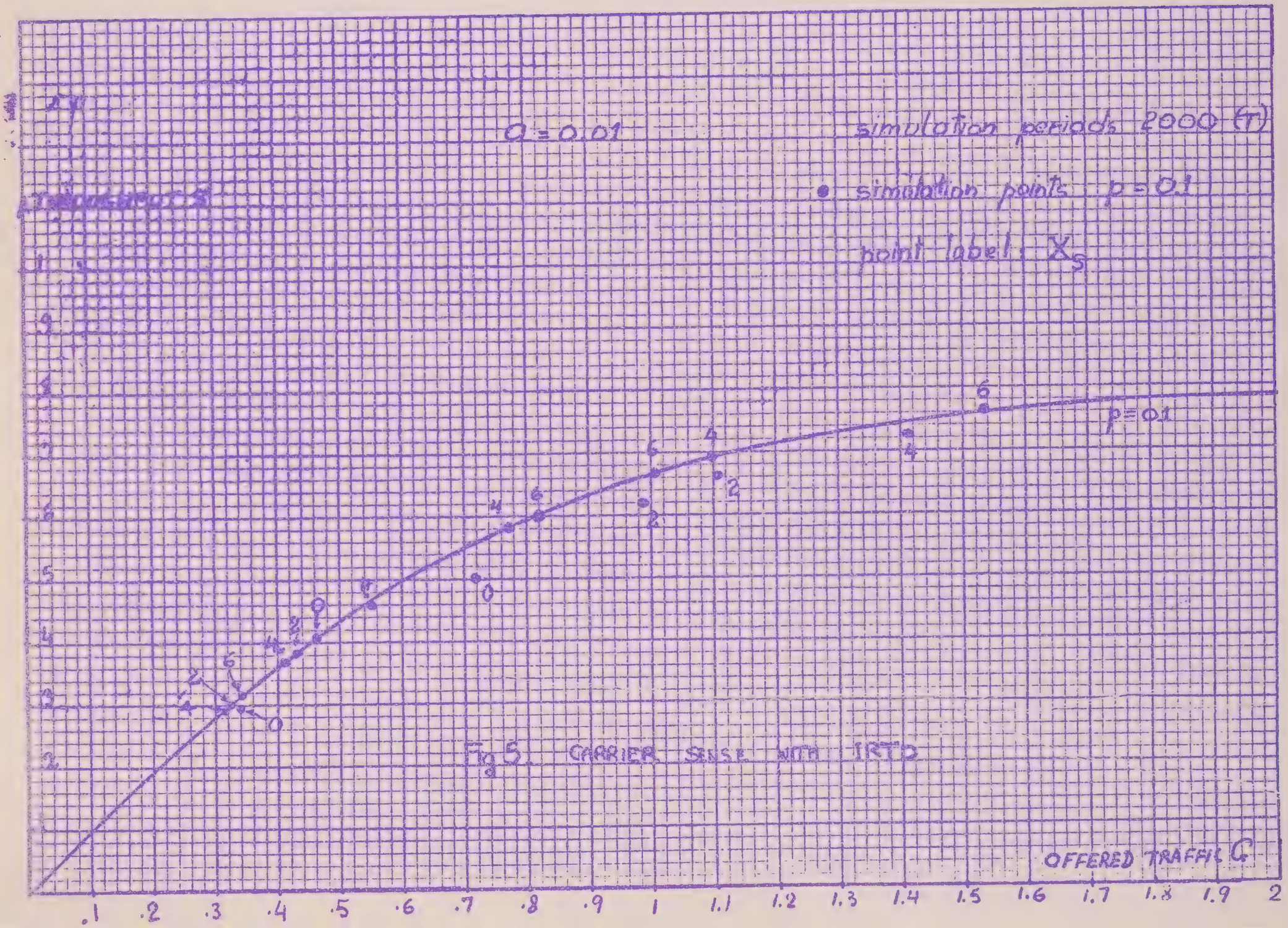


Fig 5 CARRIER SENSE WITH IRTb

K·E 5 X 5 TO ½ INCH 7 X 10 INCHES 46 0850 MADE IN U.S.A. KEUFFEL & ESSER CO.

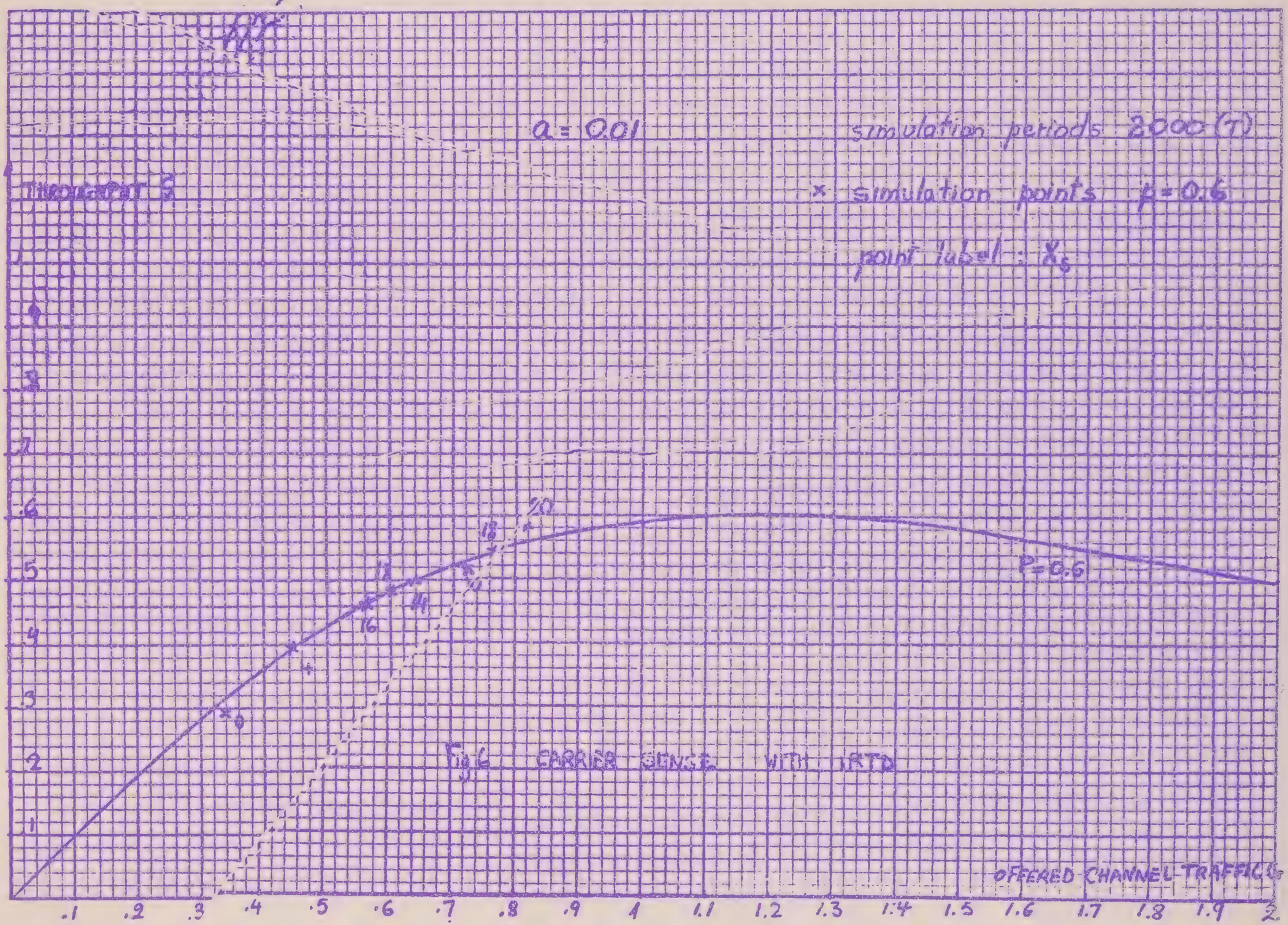


Fig 6 CARRIER SENSE WITH NRTP

K·E 10 X 10 TO THE INCH 46 0700
7 X 10 INCHES MADE IN U.S.A.
KEUFFEL & ESSER CO.

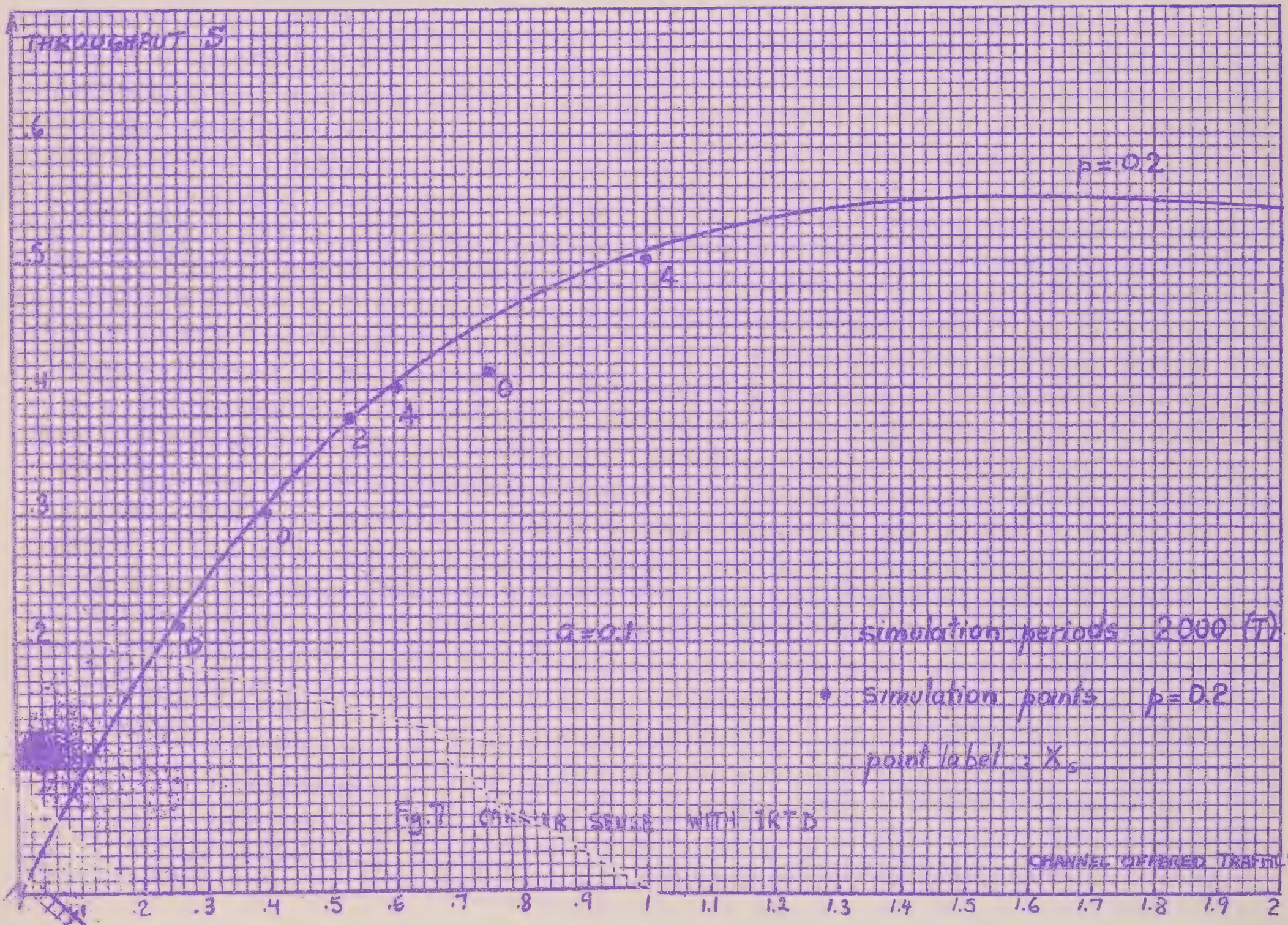
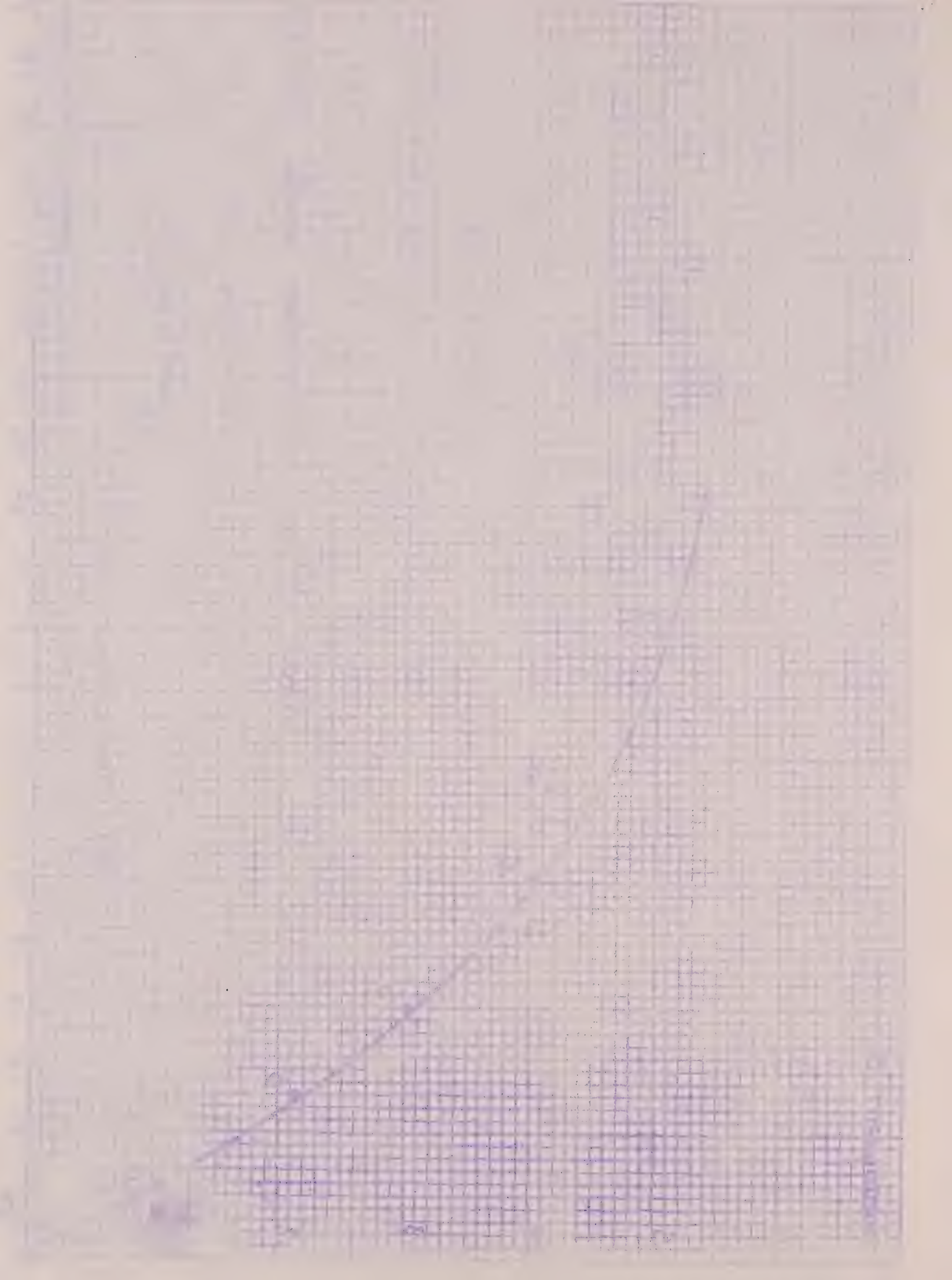
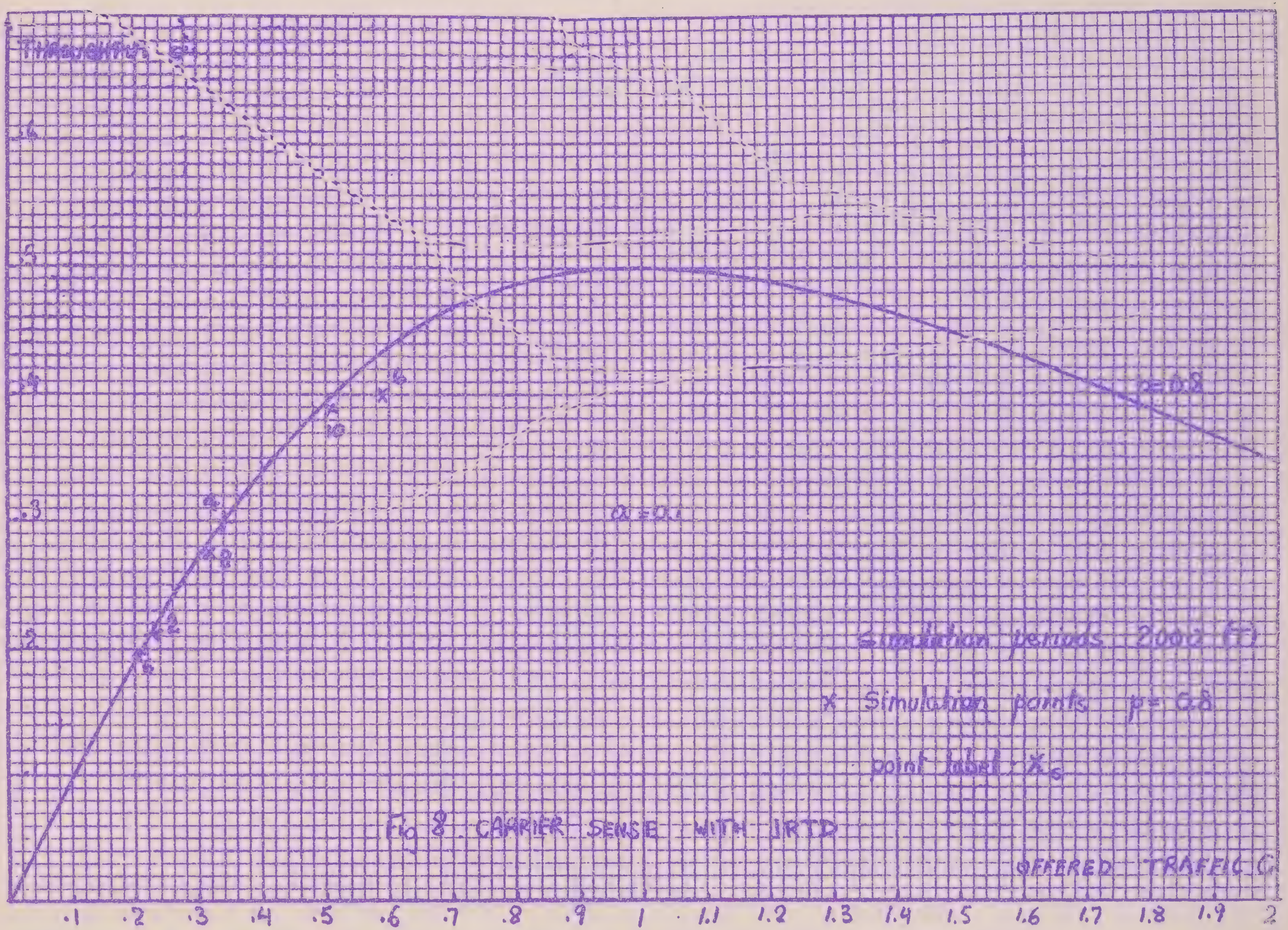
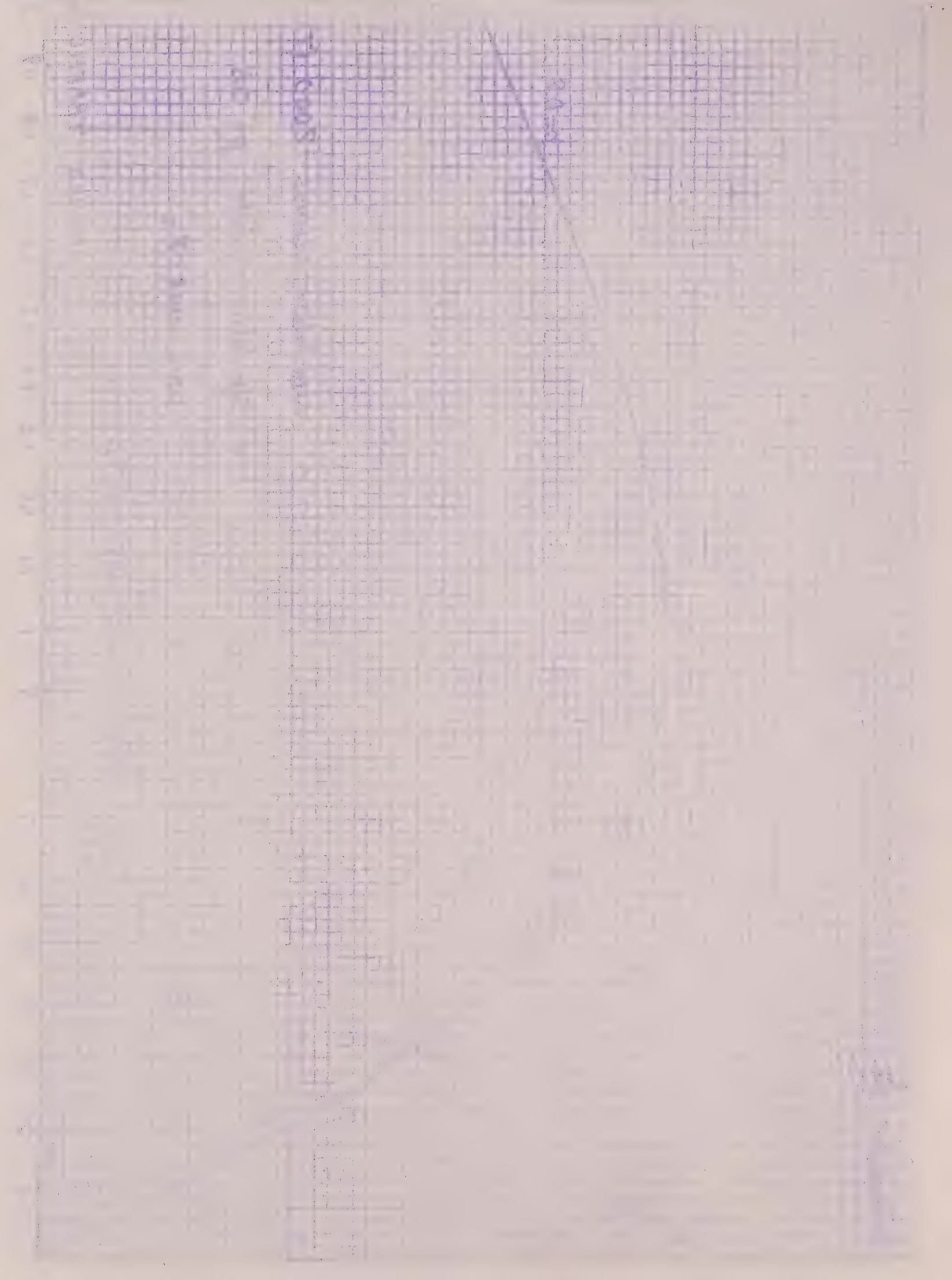


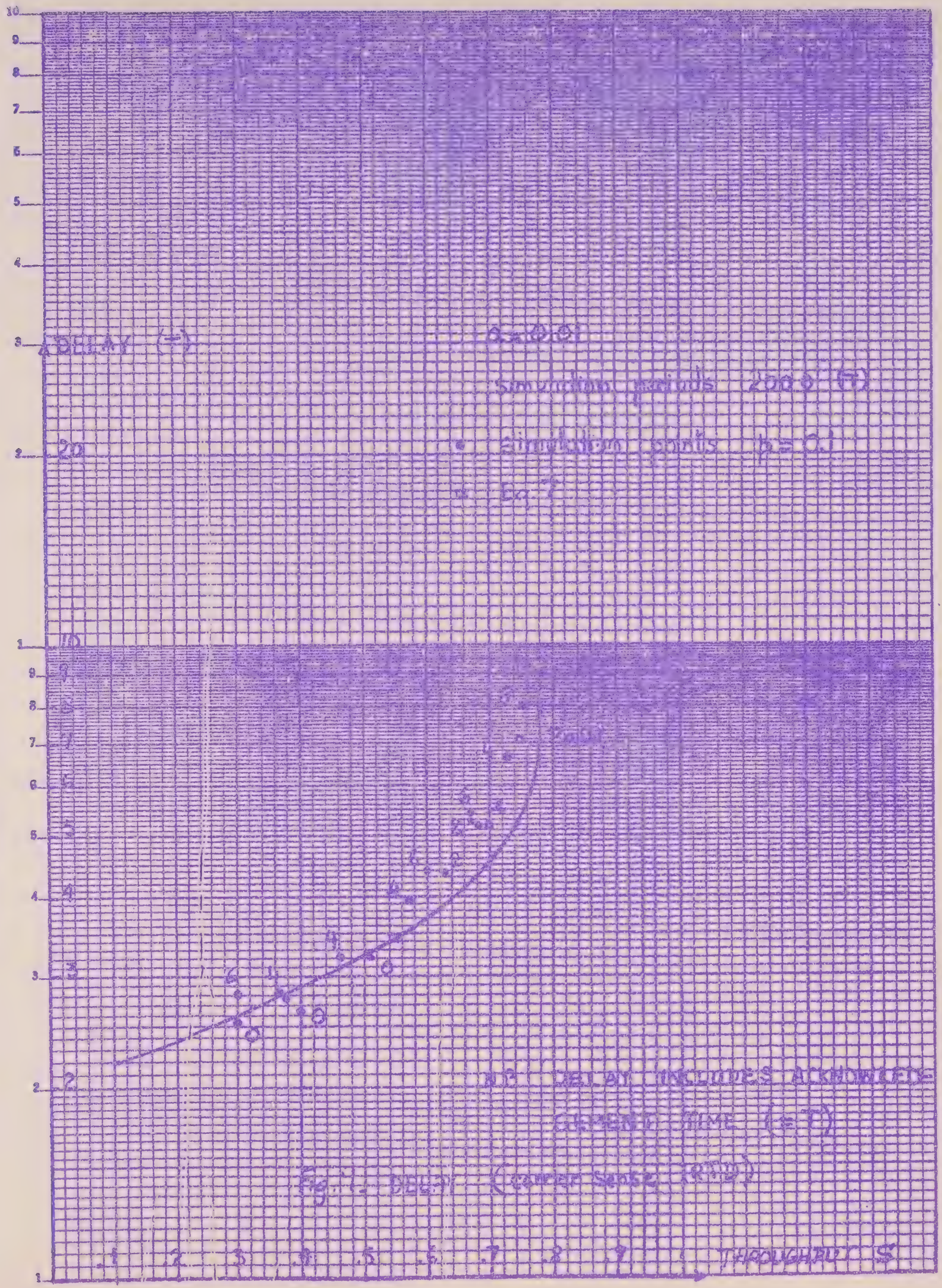
Fig.7 CARRIER SENSE WITH TRT D



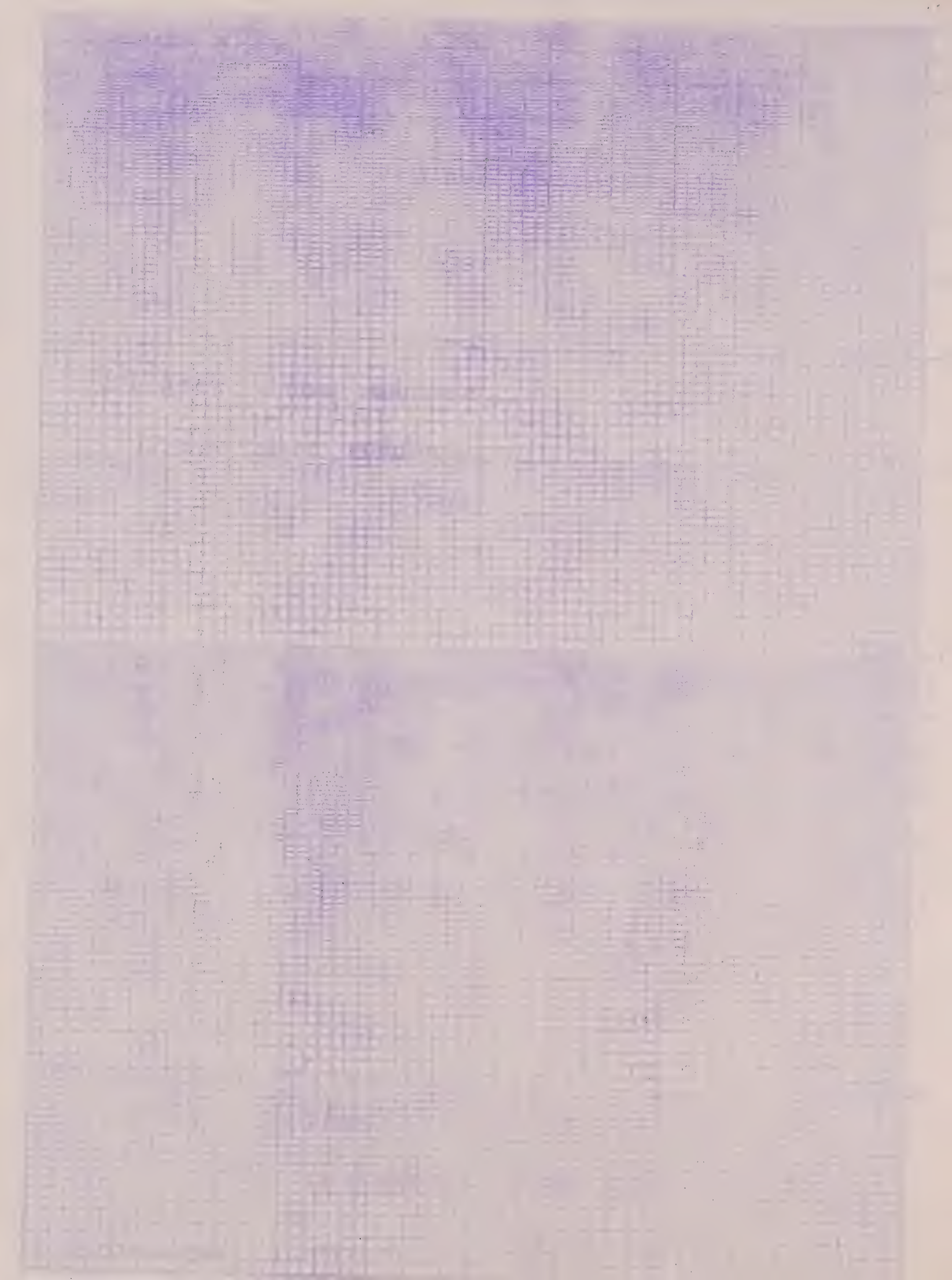
K·E 10 X 10 TO THE INCH 7 X 10 INCHES KEUFFEL & ESSER CO. 0700 MADE IN U.S.A.

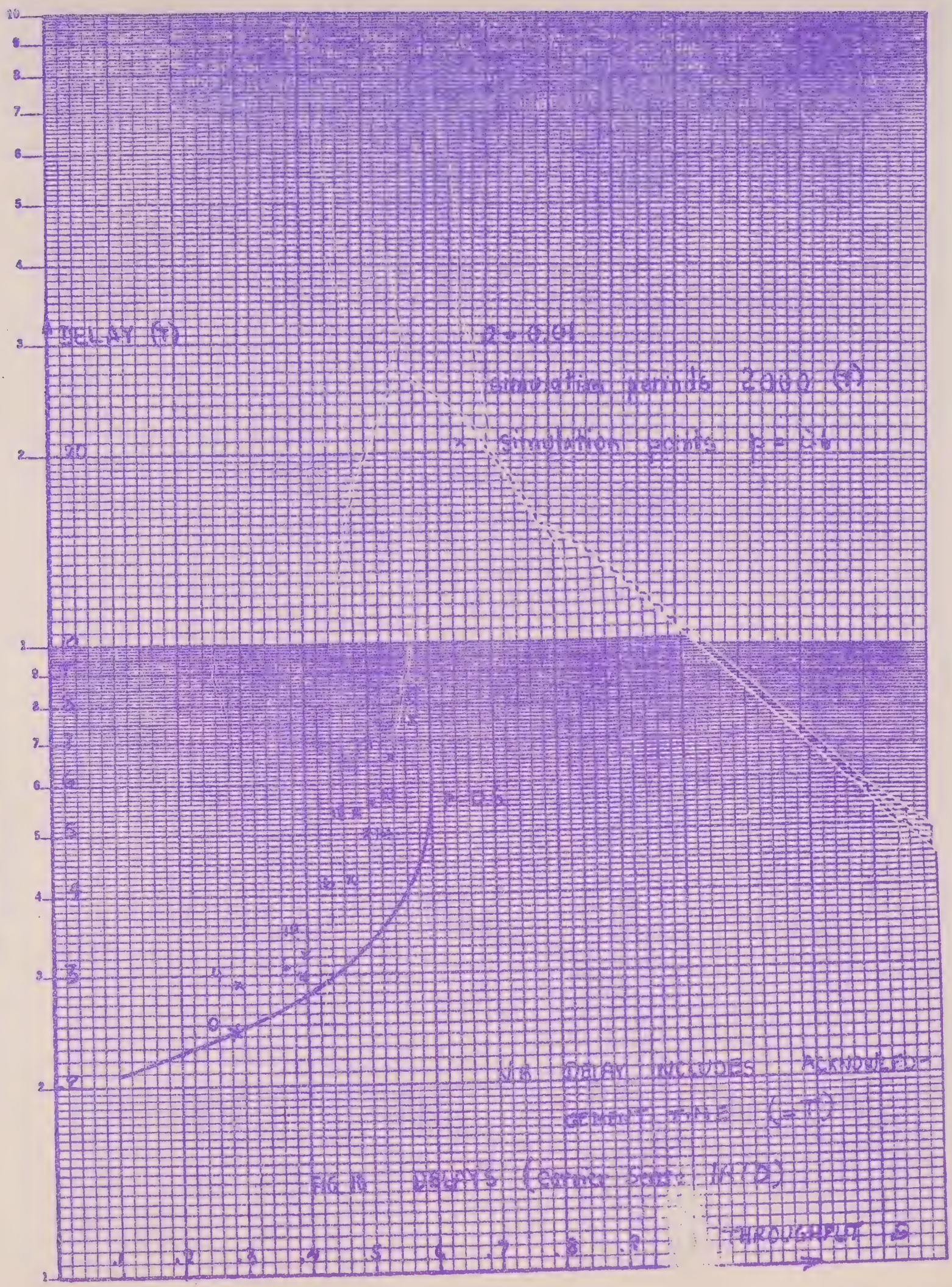




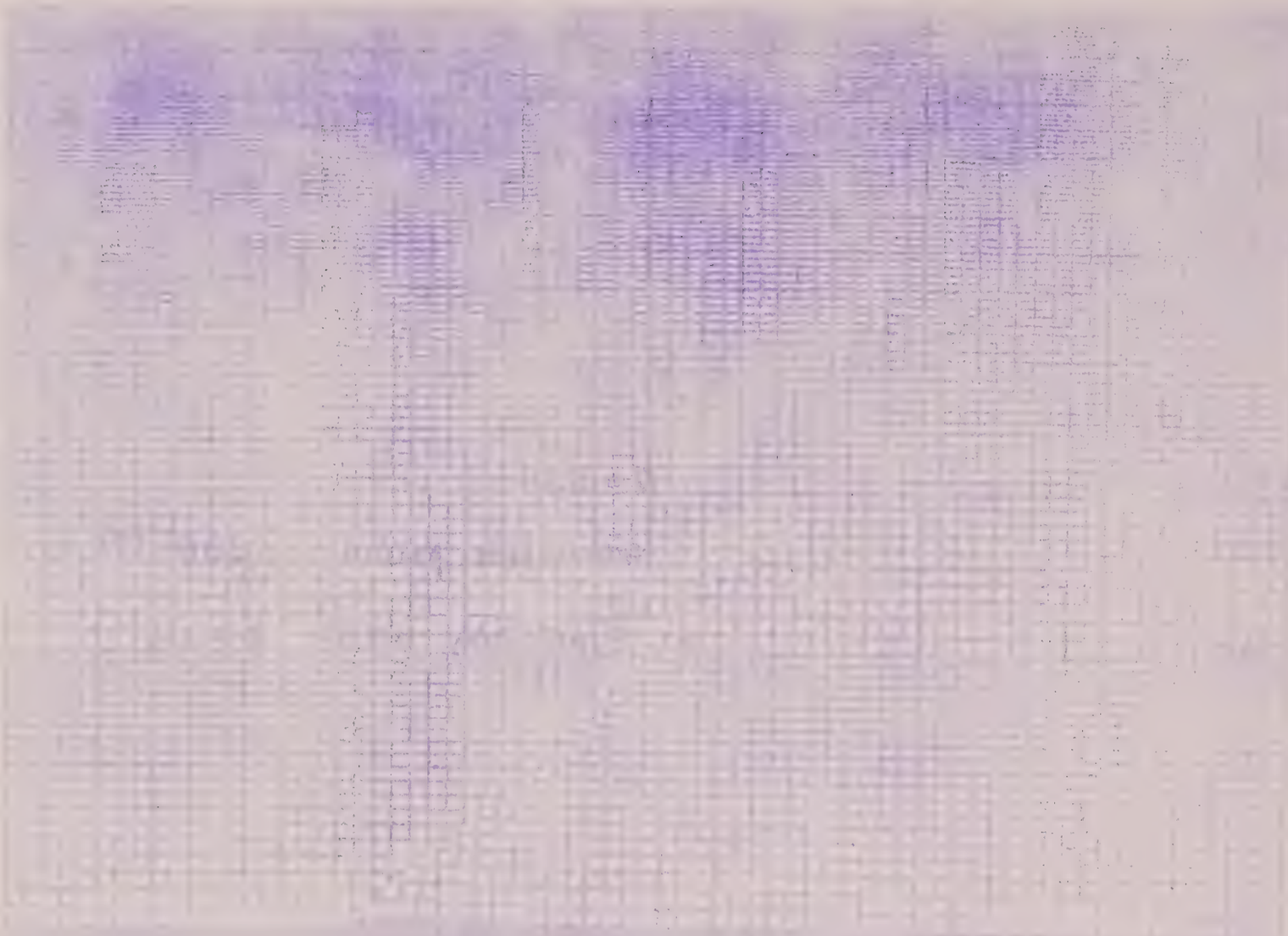


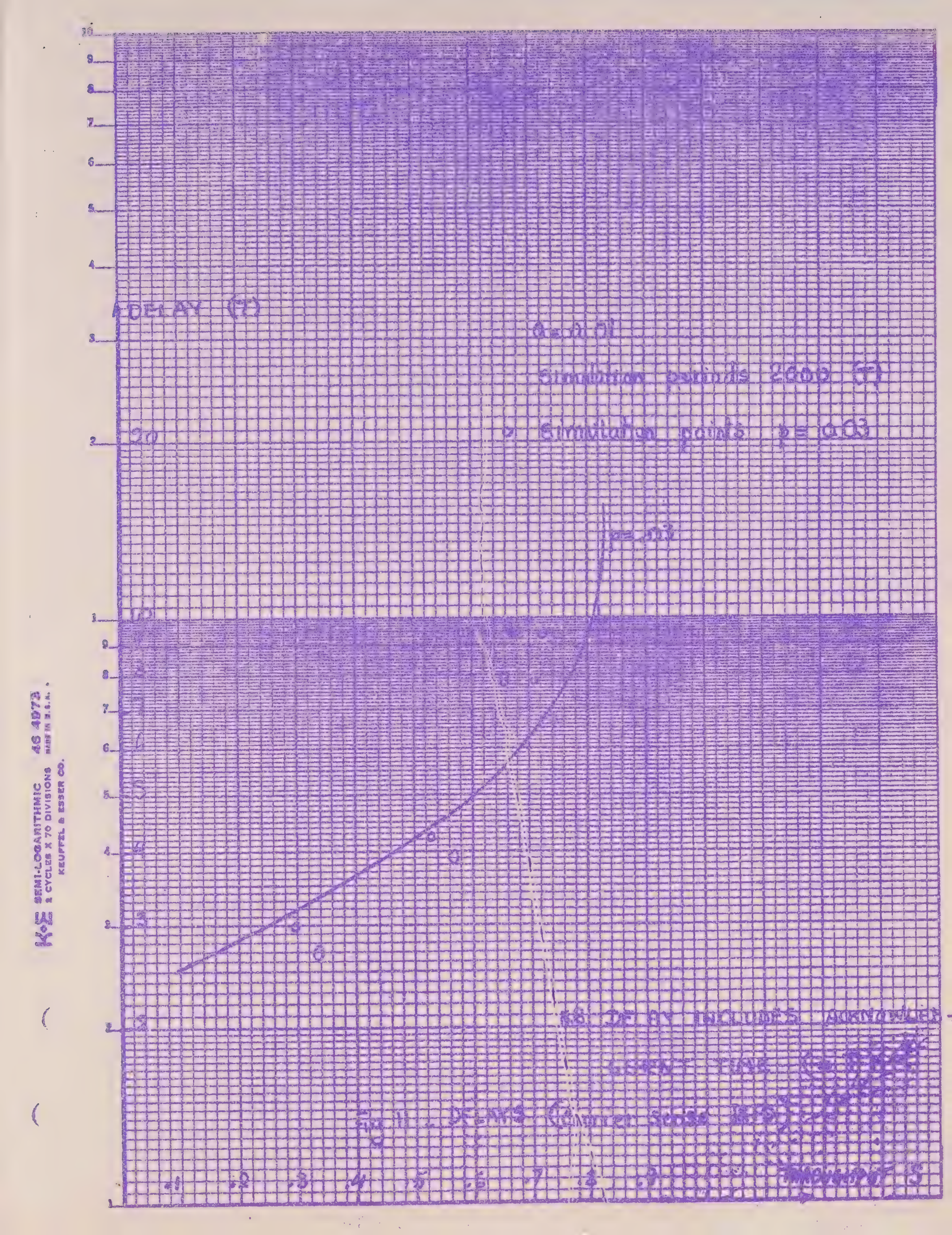
K&E SEMI-LOGARITHMIC 46 4973
2 CYCLES X 70 DIVISIONS MADE IN U.S.A.
KEUFFEL & ESSER CO.





K&E SEMI-LOGARITHMIC 46 4973
2 CYCLES X 70 DIVISIONS MADE IN U.S.A.
KEUFFEL & ESSER CO.




DELAY (T)
Simulation points 2000 (T)
Simulation points p = .03
DELAY INCLUDES ACKNOWLEDGMENT TIME
SEMI-LOGARITHMIC 46 4973
2 CYCLES X 70 DIVISIONS
KEUFFEL & ESSER CO.

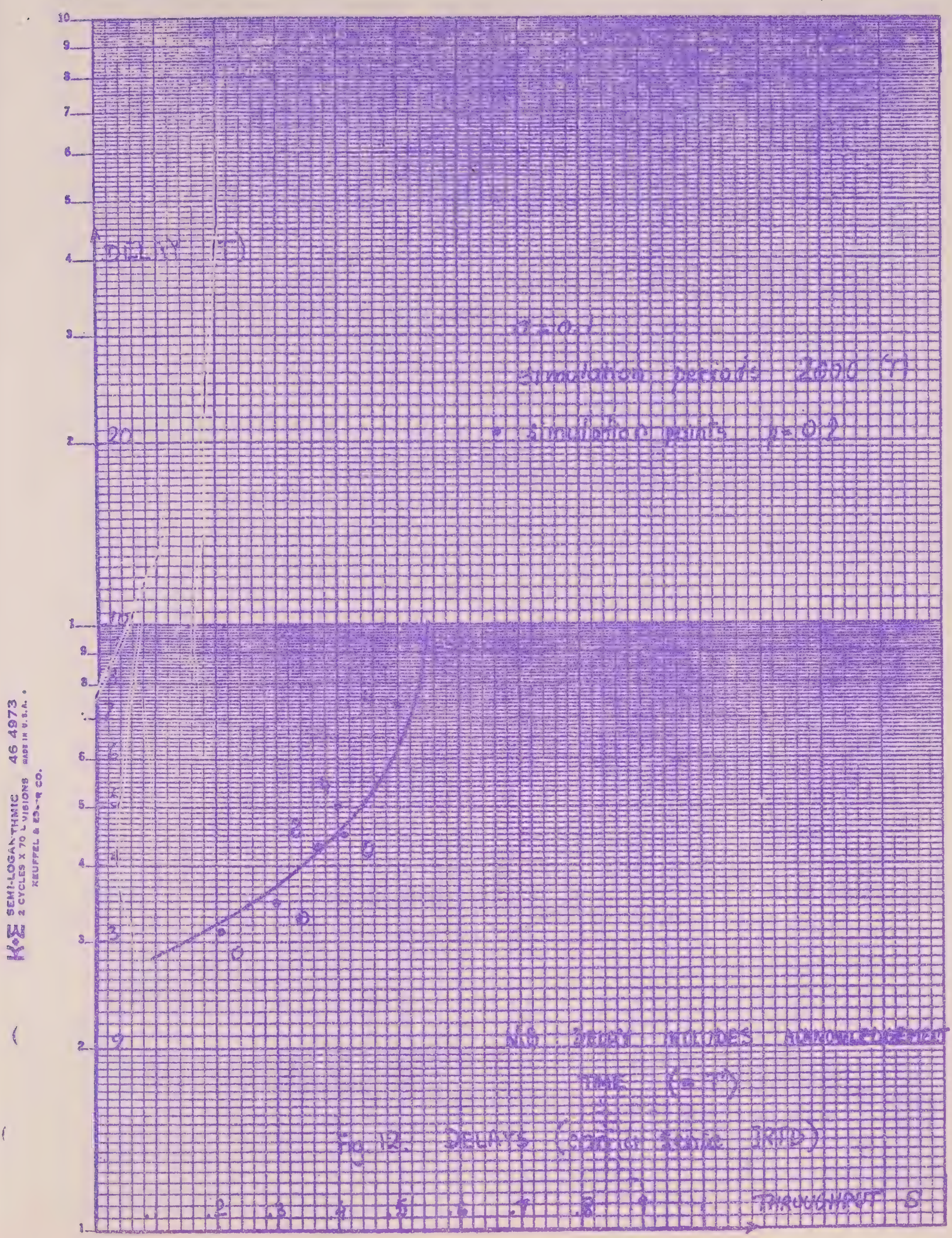
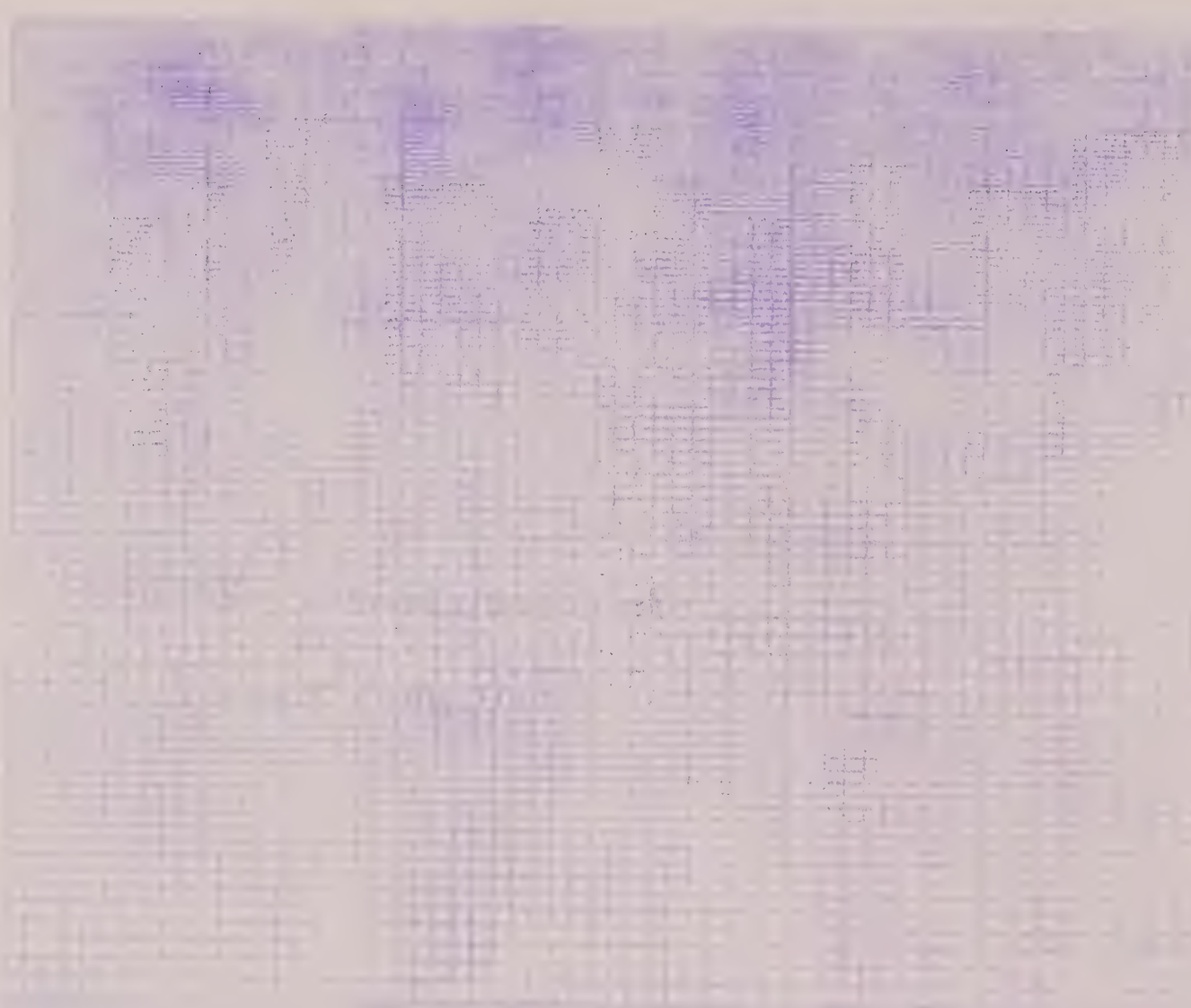
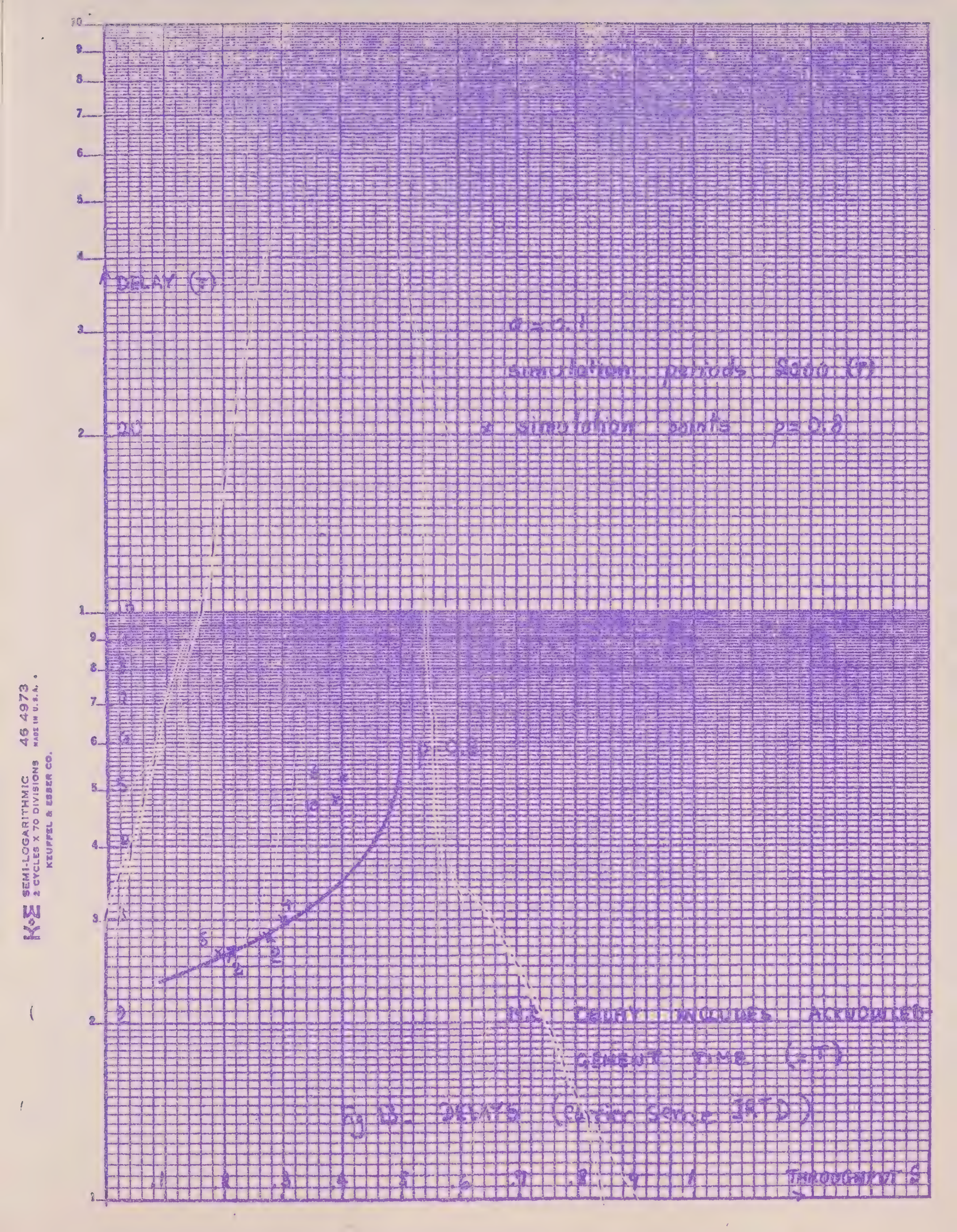


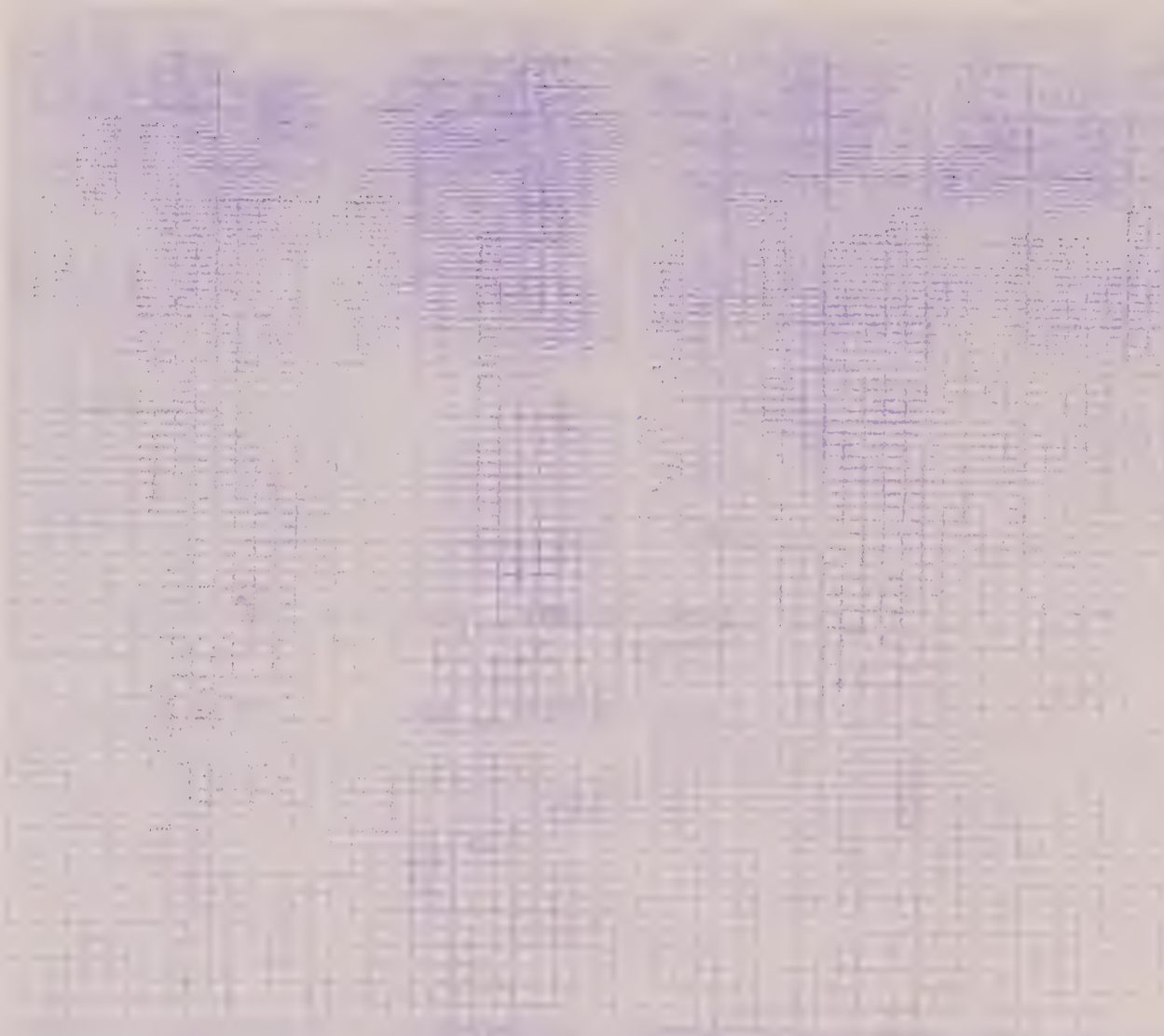
Fig. 12. DELAYS (

K+E SEMI-LOGARITHMIC 46 4973
2 CYCLES X 70 DIVISIONS MADE IN U.S.A.
KEUFFEL & ESSER CO.





K&E SEMI-LOGARITHMIC 46 4973
2 CYCLES X 70 DIVISIONS MADE IN U.S.A.
KEUFFEL & ESSER CO.



3. The Simulator

Description

The simulator is meant to be general enough to be used for different schemes with only minor changes (such as turning on or off some of the features). The following corresponds to the most general version, corresponding to carrier sense with initial random transmission delay.

The CARRIER-SENSE simulator is written in LISP 1.5, and as such is written as a series of functions. The major function in the simulator is SIMULATE. It is the principal entry point into the simulator.

Upon invoking SIMULATE the user is queried for parameters. These will be used in the simulation (they are such things as traffic rates, retransmission randomizations, and simulation durations). Once the querying is over, the simulator will announce what seed is being used for its random number generator (thus allowing experiments to be duplicated) and begins the simulation.

The network is simulated as a series of events on a time line. There are three types of events which we have named PKT, BOL and EOL respectively. Their meanings are described below.

[The time basis used in the simulation is the receiver's time axis.]

PKT. a packet enters the network.

Information contained in event:
A time (at which the event is scheduled to occur)

An initial time. (at which this packet first entered the network)
A count. (the number of times this packet has been transmitted so far)

Significance of event:
A PKT event indicates that a packet is now ready to be transmitted. Possibly it is a brand new packet entering the network, or possibly it is a packet which was unsuccessfully transmitted and is now ready to be retried.
The actual transmission time of that packet depends on TRANSDLY explained later.

BOL Beginning Of Line.

Information contained in event:
A time. (at which event is scheduled to occur)

Significance of event:
A packet was transmitted and enough time has now passed that all other nodes are aware that there is a packet being transmitted.

ECL End Of Line

Information contained in event:
same as for PKT.

Significance of event:
The transmission of a packet has been completed and enough time has elapsed so that all other nodes have seen the end of that packet. (A packet transmitted after that time can not possibly conflict with the packet who's EOL is already received).

Miscellaneous functions

There are several subfunctions in the simulator which should be briefly mentioned.

PROCESS
Called with:
An EOL event.
It assumes:
That the EOL is one for a packet which was successfully transmitted (it did not conflict with any other packets).
It does:
Extracts data from that event (delay and # of

transmissions) and updates some cumulative totals.
It returns:
nothing

DISPLAY
Called with:
nothing
It assumes:
That the simulation is ended.
It does:
Put all of the cumulative data into a pretty format and print it out on the user's console.
It returns:
nothing

TRANSDLY
Called with:
a PKT event
It assumes:
That the event represents a packet which would like to be transmitted as soon as possible.
It does:
Determine whether or not there are any reasons why that packet can't be transmitted immediately. (Packet didn't arrive on a boundary in a slotted system. There is already a packet on the line. The packet elects not to be transmitted due to the initial transmission randomization).
If packet will be delayed for one of the above reasons, TRANSDLY will reschedule the PKT event to occur after an appropriate ammount of time. (Where appropriateness is determined by the particular network under simulation).
It returns:
True, if the packet must be delayed for some reason.
False, if the packet can be transmitted immediately.

BOUNDARY
Called with:
a PKT event
It assumes:
That the event represents a packet which would like to be transmitted as soon as possible.
It does:
If the network is slotted, determine whether or not the packet entered on a slot. If it did not, BOUNDARY reschedules the PKT to occur at the next slot boundary.
It returns:
True, if the packet was rescheduled to a boundary.

```
It returns:
    True, if the problem was rescheduled to a
    slot boundary.
    False, if the problem was not moved because
    BOUNDARY requires the put to occur on a
    slot boundary, and the problem is not
    already on a slot boundary.
    False, if the network is slotted, determine
    that the problem entered on a slot
    boundary, and the problem is not
    already on a slot boundary.
    False, if the network is slotted, determine
    that the problem entered on a slot
    boundary, and the problem is not
    already on a slot boundary.
```

False, if the packet was already on a boundary or if the network is unslotted.

The Simulation

Once the network has "Warmed up", the taking of cumulative statistics is commenced. Below is the rough algorithm for the simulator.

ALGORITHM

```
LOOP:THISN <= next event on timeline;
     IF TYPE(THISN) = PKT  THEN DO;
          IF this is a brand new packet THEN
               GENERATE a new packet FOR a random time later
          /*  this is generating a new arrival according to
           the arrival process. */
          IF TRANSDLY(THISN) THEN GOTO LOOP;
          SCHEDULE a BOL event FOR tau seconds from now;
          XTIMES(THISN) <= XTIMES(THISN) + 1;
          SCHEDULE an EOL event FOR T+tau seconds from now;
          GOTO LOOP;
          END;
     IF TYPE(THISN) = BOL  THEN DO;
          LINECOUNT <= LINECOUNT + 1;
           /*  linecount is the number of packets that
             are known to be on the network at any given time.
             (a packet isn't known to be on the network until
             tau seconds after transmission commences)  */
          IF LINECOUNT > 1  THEN  ERROR <= TRUE;
               /*  error is a logical variable which is true
                if more than one packet are on the net at any
                given time.  Error remains true until all
                of the conflicting packets leave the network. */
          GOTO LOOP;
          END;
     IF TYPE(THISN) = EOL  THEN DO;
          LINECOUNT <= LINECOUNT - 1;
          IF ERROR  THEN DO;
               IF LINECOUNT = 0  THEN ERROR <= FALSE;
                 /* if the last bad packet is out of the network
                   now, the error condition is cleared.  Any packet
                   that finishes while error is true was un-
                   successful (as it conflicted with another)  */
               SCHEDULE a PKT event FOR later;
               /* this is rescheduling the packet to be
                  retransmitted (it conflicted this time)
```

```
                    exactly how long that retransmission
                    delay is is a function of the particular
                    network being simulated  */
                GOTO LOOP;
                END;
            ELSE PROCESS(THISN);
            GOTO LOOP;
            END;
        IF simulation has run long enough  THEN DO;
            DISPLAY;
            RETURN;
            END;
```

The algorithm for TRANSDLY varries from scheme to scheme but the below detailed algorithm is typical.

```
TRANSDLY:PROC(EVENT);
            IF BOUNDARY(EVENT) THEN RETURN(TRUE);
            IF LINECOUNT > 0  THEN DO;
                /*  there is already at least one packet known
                 to be on the network.  This packet cant
                 be translated until after it all clears up. */
                SCHEDULE EVENT FOR later;
                /*  exactly when is a function of the particular
                     network under simulation  */
                RETURN(TRUE);
                END;
            IF (0 < RANDOM < 1) < TRANSMISSIONPROBABILITY THEN
                RETURN(FALSE);
                /* packet can be transmitted immediately */
            ELSE DO;
                SCHEDULE EVENT FOR tau seconds later;
                /* it elected not to send, so let it try
                     again tau seconds later */
                RETURN(TRUE);
                END;
            END TRANSDLY;
```

ACKNOWLEDGEMENT

We would like to acknowledge Mark Kampe for the excellent programming work involved in this simulation.

REFERENCES

[K1] Kleinrock, L. and Tobagi, F. "Throughput in Carrier Sense (Auto Slot) Packet Radio Systems," Packet Radio Temporary Note #24, NIC 14212, February 12, 1973.

[K2] Kleinrock, L. and Tobagi, F. "Carrier Sense with Initial Random Transmission Delay," Packet Radio Temporary Note #37, NIC 15334, March, 1973.

[N1] Network Analysis Corporation, "Comparison of Hop-by-Hop and End-to-End Acknowledgement Schemes," Packet Radio Temporary Note #7, NIC 13632, January 12, 1973.